



TRABALHANDO COM FUNÇÕES DATA E HORA

DATA

Caractere de format	Descrição	Exemplo de valores retornados
<i>Dia</i>	---	---
<i>d</i>	Dia do mês, 2 dígitos com zero à esquerda	01 até 31
<i>D</i>	Uma representação textual de um dia, três letras	Mon até Sun
<i>j</i>	Dia do mês sem zero à esquerda	1 até 31
<i>l</i> ('L' minúsculo)	A representação textual completa do dia da semana	Sunday até Saturday
<i>N</i>	Representação numérica ISO-8601 do dia da semana (adicionado no PHP 5.1.0)	1 (para Segunda) até 7 (para Domingo)
<i>S</i>	Sufixo ordinal inglês para o dia do mês, 2 caracteres	st, nd, rd ou th. Funciona bem com <i>j</i>
<i>w</i>	Representação numérica do dia da semana	0 (para domingo) até 6 (para sábado)

DATA

<i>z</i>	O dia do ano (iniciando em 0)	0 até 365
<i>Semana</i>	---	---
<i>W</i>	Número do ano da semana ISO-8601, começa na Segunda (adicionado no PHP 4.1.0)	Exemplo: 42 (a 42ª semana do ano)
<i>Mês</i>	---	---
<i>F</i>	Um representação completa de um mês, como January ou March	<i>January</i> até <i>December</i>
<i>m</i>	Representação numérica de um mês, com zero à esquerda	01 a 12
<i>M</i>	Uma representação textual curta de um mês, três letras	<i>Jan</i> a <i>Dec</i>
<i>n</i>	Representação numérica de um mês, sem zero à esquerda	1 a 12
<i>t</i>	Número de dias de um dado mês	28 até 31
<i>Ano</i>	---	---
<i>L</i>	Se está em um ano bissexto	1 se está em ano bissexto, 0, caso contrário.

DATA

<code>o</code>	Número do ano ISO-8601. Este tem o mesmo valor como <code>Y</code> , exceto que se o número da semana ISO (<code>W</code>) pertence ao anterior ou próximo ano, o ano é usado ao invés. (adicionado no PHP 5.1.0)	Exemplos: 1999 ou 2003
<code>Y</code>	Uma representação de ano completa, 4 dígitos	Exemplos: 1999 ou 2003
<code>y</code>	Uma representação do ano com dois dígitos	Exemplos: 99 ou 03

HORA

<i>a</i>	Antes/Depois de meio-dia em minúsculo	<i>am or pm</i>
<i>A</i>	Antes/Depois de meio-dia em maiúsculo	<i>AM or PM</i>
<i>B</i>	Swatch Internet time	<i>000 até 999</i>
<i>g</i>	Formato 12-horas de uma hora sem zero à esquerda	<i>1 até 12</i>
<i>G</i>	Formato 24-horas de uma hora sem zero à esquerda	<i>0 até 23</i>
<i>h</i>	Formato 12-horas de uma hora com zero à esquerda	<i>01 até 12</i>
<i>H</i>	Formato 24-horas de uma hora com zero à esquerda	<i>00 até 23</i>
<i>i</i>	Minutos com zero à esquerda	<i>00 até 59</i>
<i>s</i>	Segundos, com zero à esquerda	<i>00 até 59</i>
<i>u</i>	Microssegundos (adicionado no PHP 5.2.2). Note que a função date() sempre gerará <i>000000</i> , já que aceita um parâmetro integer , enquanto que DateTime::format() possui suporte a microssegundos se DateTime foi criado com microssegundos.	Example: 654321

FUSO HORÁRIO

<i>e</i>	Identificador do fuso horário (adicionado no PHP 5.1.0)	Exemplos: <i>UTC</i> , <i>GMT</i> , <i>Atlantic/Azores</i>
<i>I</i> (i maiúsculo)	Se a data está ou não no horário de verão	1 se horário de verão, 0, caso contrário.
<i>O</i>	Deslocamento ao Horário de Greenwich (GMT) em horas	Exemplo: +0200
<i>P</i>	Deslocamento ao Horário de Greenwich (GMT) com dois pontos entre horas e minutos (adicionado no PHP 5.1.3)	Exemplo: +02:00
<i>T</i>	Abreviação do fuso horário	Exemplos: <i>EST</i> , <i>MDT</i> ...
<i>Z</i>	Deslocamento, em segundos, do fuso horário. O deslocamento para fusos horários a oeste de UTC sempre será negativa, e para aqueles à leste de UTC sempre será positiva.	-43200 até 50400

DATA E HORA COMPLETA

<i>c</i>	Data ISO 8601 (adicionado no PHP 5)	2004-02-12T15:19:21+00:00
<i>r</i>	» RFC 2822 formatted date	Exemplo: <i>Thu, 21 Dec 2000 16:01:07 +0200</i>
<i>U</i>	Segundos desde Unix Epoch (January 1 1970 00:00:00 GMT)	Veja também time()

DATA DE HOJE

```
<?php  
echo "Data atual: " . date("Y-m-d");  
?>
```

OBTER A HORA ATUAL:

```
<?php  
echo "Hora atual: " . date("H:i:s");  
?>
```

OBTER A DATA E HORA ATUAL COMPLETA:

```
<?php  
echo "Data e hora atuais: " . date("Y-m-d H:i:s");  
?>
```

FORMATAR UMA DATA ESPECÍFICA:

```
<?php
$data = "2024-06-02";
$data_formatada = date("d/m/Y", strtotime($data));
echo "Data formatada: " . $data_formatada;
?>
```

ADICIONAR/SUBTRAIR DIAS A PARTIR DE UMA DATA:

```
<?php
$data = "2024-06-02";
$nova_data = date("Y-m-d", strtotime($data . " +1 day"));
echo "Nova data: " . $nova_data;
?>
```

ADICIONAR/SUBTRAIR DIAS A PARTIR DE UMA DATA:

```
<?php
$data1 = "2024-06-02";
$data2 = "2024-06-10";
$diferenca = strtotime($data2) - strtotime($data1);
$dias = floor($diferenca / (60 * 60 * 24));
echo "Diferença em dias: " . $dias;
?>
```

FUNÇÃO DATE()

date(): Esta função é usada para formatar uma data e/ou hora de acordo com um formato especificado.

```
<?php  
echo date("Y-m-d H:i:s"); // Saída: 2024-06-02 14:30:45  
?>
```

FUNÇÃO TIME()

time(): Retorna a data atual em segundos

```
<?php  
echo time(); // Saída: 1622624473 (um número que representa os segundos desde a Época  
?>
```

FUNÇÃO GETDATE():

getdate(): Retorna um array associativo contendo informações sobre a data/hora fornecida ou a data/hora atual, se não for fornecida.

```
<?php
$data = getdate();
print_r($data);
/*
Saída (exemplo):
Array
(
    [seconds] => 45
    [minutes] => 30
```



```
[hours]    => 14
[mday]     => 2
[wday]     => 5
[mon]      => 6
[year]     => 2024
[yday]     => 153
[weekday]  => Friday
[month]    => June
[0]        => 1622625045

)
*/
?>
```

FUNÇÃO MKTIME()

A função `mktime()` retorna o timestamp Unix para uma data/hora específica.

Um **timestamp** (ou marca temporal) é um número inteiro que representa o número de segundos desde o início da **Era Unix** (também conhecido como o Epoch Unix), que começou à meia-noite de 1º de janeiro de 1970, no horário UTC (Tempo Universal Coordenado). Esse sistema de contagem é amplamente utilizado em sistemas de computadores e linguagens de programação para rastrear e armazenar informações de data e hora de maneira uniforme.

FUNÇÃO MKTIME()

```
<?php
// mktime(hour, minute, second, month, day, year)
$timestamp = mktime(14, 30, 0, 6, 14, 2024);
echo "Timestamp específico: " . $timestamp . "\n";
echo "Data formatada: " . date("Y-m-d H:i:s", $timestamp) . "\n";
?>
```

FUSO HORÁRIO

Para trabalhar com fusos horários, você pode usar a classe **DateTimeZone**.

ÁfricaAfrica/Cairo

Africa/Johannesburg

Africa/Lagos

Africa/Nairobi

FUSO HORÁRIO

América

America/Argentina/Buenos_Aires

America/Chicago

America/Los_Angeles

America/Mexico_City

America/New_York

America/Sao_Paulo

FUSO HORÁRIO

Ásia

Asia/Bangkok

Asia/Dubai

Asia/Hong_Kong

Asia/Kolkata

Asia/Shanghai

Asia/Tokyo

FUSO HORÁRIO

Europa

Europe/Berlin

Europe/Lisbon

Europe/London

Europe/Madrid

Europe/Paris

Europe/Rome

FUSO HORÁRIO

Oceania

Australia/Sydney

Pacific/Auckland

Pacific/Fiji

FUSO HORÁRIO

```
<?php
$date = new DateTime("now", new DateTimeZone("America/Sao_Paulo"));
echo "Data e hora em São Paulo: " . $date->format("Y-m-d H:i:s") . "\n";

$date->setTimezone(new DateTimeZone("Europe/London"));
echo "Data e hora em Londres: " . $date->format("Y-m-d H:i:s") . "\n";
?>
```

FUNÇÃO STRTOTIME

Serve para fazer cálculos com data.

Ex:

```
date_default_timezone_set('America/Sao_Paulo');  
echo date("d/m/Y", strtotime("now")); //data atual  
echo "<br />";  
echo date("d/m/Y", strtotime("+1 day")); //data atual mais 1  
dia  
echo "<br />";
```

FUNÇÃO STRTOTIME

```
echo date("d/m/Y",strtotime("+1 week")); //data atual mais  
uma semana  
echo "<br />";  
echo date("d/m/Y",strtotime("next Tuesday")); // próxima  
terça feira
```

FUNÇÃO STRTOTIME

```
echo "<br />";  
echo date("d/m/Y",strtotime("+1 month")); // acrescenta um  
mes  
echo "<br />";  
echo date("d/m/Y",strtotime("last tuesday")); //última terça  
feira  
echo "<br />";
```